



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/449,021	11/24/1999	HELMUT EMMELMANN	EMME-1000US0	5718
8685	7590	04/04/2012	EXAMINER	
DERGOSITS & NOAH LLP			KENDALL, CHUCK O	
Three Embarcadero Center				
Suite 410			ART UNIT	PAPER NUMBER
SAN FRANCISCO, CA 94111			2192	
			MAIL DATE	DELIVERY MODE
			04/04/2012	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

1 RECORD OF ORAL HEARING
2 UNITED STATES PATENT AND TRADEMARK OFFICE
3

4 BEFORE THE BOARD OF PATENT APPEALS
5 AND INTERFERENCES
6

7 *Ex Parte HELMUT EMMELMANN*
8

9 Appeal 2012-001997
10 Application 09/449,021
11 Technology Center 2100
12

13 Oral Hearing Held: February 14, 2012
14

14 Before MARC S. HOFF, CARLA M. KRIVAK, and THOMAS S. HAHN,
15 *Administrative Patent Judges.*

16 APPEARANCES:

17 ON BEHALF OF THE APPELLANT:

18 MICHAEL E. DERGOSITS, ESQUIRE
19 Dergosits & Noah, LLP
20 Three Embarcadero Center, Suite 410
21 San Francisco, California 94111
22
23
24
25
26

1

2

The above-entitled matter came on for hearing on Tuesday,
February 14, 2012, commencing at 1:41 p.m., at the U.S. Patent and
Trademark Office, 600 Dulany Street, Alexandria, Virginia, before Tania
Kendall, a Notary Public.

3

4

5

P R O C E E D I N G S

6

THE USHER: Calendar No. 36, Appeal No. 2012-0011997,
Mr. Dergosits.

7

8

JUDGE HOFF: Thank you. Welcome.

9

MR. DERGOSITS: Thank you.

10

JUDGE HOFF: Good afternoon. Do you have a business card for the
Reporter, or can you spell your name? You have 20 minutes. You may
begin when ready.

11

MR. DERGOSITS: Good afternoon. My name is Michael Dergosits.
Today, I have with me Mr. Richard Knapp (ph), a Patent Attorney who's
been prosecuting the case, and Appellant Emmelmann who is the Inventor,
who's come from Germany to participate in the hearing today.

12

The place that I wanted to begin today was to talk a little bit about the
procedural place where we find ourselves because it's a little bit unusual.
This application and the claims that are at issue in the appeal today were
subject to a Notice of Allowance in 2008. And in 2008 after the Notice of
Allowance was issued, the matter was referred to Quality Review. And at
that time, the Notice of Allowance was withdrawn by Quality Review, and
the reason or rationale, apparently, for the withdrawal from issuance was for
the Examiner to cite and consider the Web Writer art that's of record today.

13

1 The Examiner ultimately issues a rejection based on Web Writer I and
2 Web Writer II, the two primary references at issue even today. And
3 ultimately, the Examiner issued a final rejection based on those two
4 references. A Notice of Appeal was filed and an opening Appeal Brief was
5 filed. An Examiner's Answer was filed. A Reply Brief was filed. And after
6 the Appellant's first Reply Brief, in a phone conversation with the Examiner
7 and his SBE, it was suggested that there needed to be the citation of an
8 additional reference, the Faustini (ph) reference, which is of record in the
9 matter today. And so, over the course of several weeks, after the first reply
10 brief was filed, Mr. Knapp and Mr. Emmelmann and the Examiner and the
11 SBEs attempted to work out some resolution of how Faustini would be cited
12 and whether there were amendments or arguments that would permit the
13 allowance of the pending claims at the time.

14 Those discussions didn't succeed in reaching an agreement. And so,
15 the Examiner then filed a second Examiner's Answer in which the original
16 final rejection, based solely on the two Web Writer articles was withdrawn,
17 and, in its place, was substituted a new rejection, which takes in some claims
18 Web Writer I and combines it with the Faustini article, and then other
19 rejections of the claims cites the Web Writer II article and combines it with
20 Fasteny.

21 So I thought it was important, with all of that background, to describe
22 the reason that we're here today, and the situation where this is the first time
23 that the rejection is being argued at all. So with that in mind, I wanted to
24 discuss what this invention was about and how it, then, tracks down to the
25 claims in issue.

26

1 Again, this application's been pending for quite a long time. So this
2 invention was made and filed in 1999. And at the time, what we're talking
3 about is a web application where the web application is made up of
4 document templates. The document templates contain both text and
5 instructions. And there is an application which runs the template to create a
6 generated document. When there is a web application, there's a client
7 computer that's running a Internet browser. And the browser issues a
8 request for a page. And in response to the request for a page, the web
9 application generates the page and sends it through the browser for display.

10 In that context, or in that environment, this invention relates to the
11 ability to edit document templates so that one could change the page that's
12 displayed on the browser. The problem with what existed at the time that
13 this invention was made is that the document template that's used to generate
14 the page that's shown on the browser doesn't look like the page that's
15 generated for display on the browser.

16 It's a series of text and computer instructions. And so, if one wanted
17 to edit a displayed page, you had to work on the document template. And if
18 you made changes to the document template, it needed to be processed by
19 the web application in order to then generate another page for display so you
20 could see what you had done. And so, this sequence of events of editing the
21 template and generating a new document and sending that document for
22 display on the Internet browser was something that existed, but it was
23 problematic. One was that it required multiple steps and additional time to
24 edit the document, save it, send it to the web application, run the web
25 application, create the new page, and then view the page. So, that process
26 was lengthy and multiple steps.

1 The other problem was, in this environment that existed before the
2 invention, is that if you located an error in the display, the web page that you
3 were looking at on your Internet browser, if you saw that there was
4 something that was off, it was difficult to locate the place in the document
5 template that generated that error. So even if you could see what was going
6 on, then tracing that back to the location in the template where that error
7 occurred, it was tedious, I guess. It's not difficult, but it's tedious.

8 So this invention solved this problem. And the way that this invention
9 solves the problem is to enable the developer to see the page that would be
10 on the browser display, with editing features on that page, which enabled the
11 developer to go into the document template at the location corresponding to
12 the display, making the edits. And then, this editor program generates a new
13 page with the edited template. And the developer can immediately see the
14 results of the editing that took place.

15 Now, one thing that the Web Writer art brings in and which is not
16 surprising at all to us is that the concept of document generation from a
17 template was not a paradigm or a development or innovation that
18 Dr. Emmelmann created. That was the paradigm for generating a stack of
19 document templates that when run by the application, generated a web page.

20 What innovation is present in the editor that's in the claims here is that
21 the displayed page that comes to the browser has editing features on it so
22 that when the developer is looking at the display in the browser, they can use
23 a click of a mouse to open up the document template that corresponds to that
24 portion of the display.

25 So one of the things that we will talk about, with respect to the claim
26 specifically is that there's a document-generator program. That, by itself,

1 wouldn't be a patentable advance. But there's a document-generator
2 program that adds editing features to the final generated document. So that's
3 the place where we're starting from. The advantages that flow from this
4 invention is that the developer can see the changes instantaneously. And the
5 developer can locate the location in the template where the changes need to
6 be made, without having to do the tedious work that existed in the prior art.

7 Now, of course, what matters to all of us is what the claim language
8 says. And so, the first independent claim that's at issue is Claim number 1.
9 And it's a computer-readable medium claim, computer programs having
10 executable instructions for executing software applications. And the
11 software applications run on a data network, coupling a server computer and
12 a client computer.

13 The client computer runs a browser program. And where, upon
14 request by the browser program, at least one of the applications generates
15 generated documents for display by the browser program on a display
16 device, and responds to the request for the generated documents. So the
17 browser issues a request for a page that they want to edit and the computer
18 program responds with a generated document. So the instructions that are
19 on the computer-readable medium are a document-generator program
20 running at least part of one of the applications being edited, and generating
21 the generated documents; the generated documents, including additional
22 editing features, for interpretation by the browser program.

23 The second instruction set on the medium is an editor program that
24 dynamically operates on the generated documents displayed by the browser
25 program, via the editing features. Now, one of the things that presented a
26 challenge in the development of this invention was that document templates

1 can all include aspects that are to be displayed that are static, meaning that
2 they're text or an image that doesn't change. The document template can
3 also include executable instructions, or scripts, that have to be run in order to
4 generate information or fields or images. And so, what is difficult and what
5 was the reason why people worked on document templates in the past is if
6 the editor is not running the application, then the editor can't execute the
7 scripts or other instructions, and therefore provide a display of the active or
8 dynamic parts of the document template. By generating the document
9 outside of the browser, a document can be generated with the full
10 capabilities of the application under which the template was developed. If
11 you do it inside the browser, then the browser doesn't necessarily have all of
12 the capability that's necessary to run the document template, scripts, or code.

13 The next independent claim that's at issue is Claim 22. And that is
14 also a computer-readable medium claim for computer programs that edit and
15 maintain applications using web browser, and comprises an editor program
16 operating within the web browser on generated documents; having
17 instructions for inserting, deleting, and modifying components on the
18 document templates. And there's a document-generator program having
19 instructions for processing document templates, for executing said
20 components, and generating the generated documents for the document
21 templates, in a manner that's understandable by the web browser.

22 Claim 26 is a system claim. The limitations focus on the server
23 computer, what instructions the server computer is processing. And so, you
24 have a document store, which is where the templates are maintained.
25 The first software program, including instructions for transforming at least
26 one first document retrieved from the document store, into a second

1 document, having features which permit editing of the first document. So
2 this relates to the document generation, taking a template, which would be
3 the first document, and transforming it into a generated page, being the
4 second document, and having the features which permit editing of the
5 template through the generated document. A second software program,
6 including instructions to receive information from the client computer and
7 instructions to modify the first document stored at the document store, that's
8 the editor capability.

9 Claim 59 is the other independent claim. But in the interest of -- I
10 thought the limitations were, well, similar to the ones that we discussed, in
11 terms of the scope. They're different words, but the scope is similar.

12 So I thought I would talk a little bit about the rejections of record.
13 One of the things, I think, that's noteworthy here is that the final rejection for
14 which the Notice of Appeal was originally filed was based on the Web
15 Writer articles, alone. And the arguments that were made to this Board in
16 the opening Appellant's brief is that the Web Writer editor program doesn't
17 contain a capability for document generation within the editor program.
18 And so, without a document-generator program within the editor program,
19 there's no generation of documents. And therefore, it doesn't have the
20 capability that the claim requires, which is to generate a document with
21 editing features on the generated document.

22 One side issue is that Web Writer also had another module, which was
23 a page-generator module. That page-generator module was not affiliated or
24 connected in any way to the editor capabilities of page writer. And so, the
25 page writer in the Web Writer prior art is the same as a page generator or
26 document generator in any web application.

1 The Web Writer page generator, or document generator, doesn't
2 provide anything new or special. But ultimately, it appears as if the
3 Examiner agreed that the Web Writer editing function didn't meet the claim
4 limitations, with respect to the independent claims that I just recited, and the
5 claims that are pending that are dependent from those, and withdrew a final
6 rejection based on that part alone.

7 So what happened subsequent to the withdrawal of the rejection is the
8 statement of a new grounds for rejection, which brings in a new piece of
9 prior art called Faustini. And what I'd like to do is spend a few minutes to
10 talk about Faustini because that's what the new issue is here today.

11 The Examiner citation of Faustini appears to relate to one topic. And
12 that has to do with the ability to edit and display what were referred to as
13 dynamic content, all right? So I just said earlier one of the problems with
14 editing a document template is that there are aspects of the template that are
15 active, or dynamic, which require it to be run. And so, we had made the
16 argument that Web Writer didn't provide the ability to edit those active
17 portions and display them, at the same time, in the editor. And so, the
18 Examiner went outside of Web Writer to cite Faustini, purportedly for the
19 concept that it was possible to edit dynamic content.

20 JUDGE HOFF: Sorry about that, please continue?

21 MR. DERGOSITS: What I'd like to point out about Faustini is,
22 number 1, it comes from an area of development that is different than the
23 web application that the present invention relates to, and that the Web Writer
24 articles come from. And those are web applications which involve
25 document templates that are run by the application to create a web page,
26 displayed by a browser. Faustini doesn't follow that paradigm. Faustini uses

1 java applets. And if you look at the Web Writer II prior art, there's a section
2 in the beginning of it called related work. And in that section of the Web
3 Writer II article, they talk about, there are two patents, essentially.

4 One of them is the paradigm that they're working in, which is
5 document templates that run to make web pages. Or there are things that
6 could be run on the client, involving applets or java programming. And so,
7 it's clear that Faustini falls into a different category than the one that Web
8 Writer is working in.

9 The second thing is that Faustini is discussing and talking about
10 applets that run on the client's side, within the web browser. And the kind of
11 processing and the kind of application running that we're talking about in the
12 claims, as document generation, is something that takes place outside of the
13 browser. And so, even though Faustini does describe and discuss and
14 illustrate that there is dynamic content, the dynamic content that is discussed
15 and taught by Faustini isn't an application that's outside of the browser. And
16 so, for the Examiner's rejection, he needs to support the concept that Faustini
17 would have taught how to modify Web Writer to provide the capability to
18 edit dynamic content and display dynamic content within the editor. And we
19 would submit that that is not anything that Faustini shows and that Web
20 Writer itself just shows that that's a different approach than what is
21 considered in Web Writer, and what's the subject of this application.

22 JUDGE HOFF: Counsel, your time is essentially expired, but I have a
23 couple of questions for you. As you just said, it's your position that Faustini
24 teaches this client-side processing, whereas your position is your invention is
25 directed to its server side execution. What I'm getting at is, can you express
26 for me how that feature is brought out in each of your independent claims?

1 MR. DERGOSITS: Well, more importantly, I don't think that
2 Faustini uses the paradigm of templates generating documents. And so,
3 absent templates generating documents, then whatever's taught in Faustini
4 about executing and editing client-side software is interesting but not
5 relevant to the problem that we were trying to solve.

6 JUDGE HOFF: And by application, you're asserting that Web Writer
7 doesn't teach that, either?

8 MR. DERGOSITS: Yes. That's correct. I don't think that there's any
9 question that Web Writer, by itself, doesn't teach the ability to edit dynamic
10 aspects of the document template, and display the dynamic aspects in the
11 editor program.

12 JUDGE HOFF: And by dynamic, you're talking about the back-and-
13 forth about editing running applications?

14 MR. DERGOSITS: That's correct.

15 JUDGE HOFF: My second question has to do with the word similar
16 in, for example, Claims 26 and 59. Can you give us any guidance on how
17 we might construe the term similar, for example, in 59, dynamic web
18 documents which look and function similar to the end user's view of the
19 documents? If I take that on its own, without context, the word similar is
20 open to substantial interpretation.

21 MR. DERGOSITS: Well, I guess the concept would be that it looks
22 very much like, but doesn't have to be an exact replica. There may be some
23 changes to it.

24 JUDGE HOFF: With regard to the language in Claim 26, which says,
25 for example, such that at least a part of the second document appears and
26 functions similar to the run-time view of the first document, is there some

1 reason that we should not construe the part that appears and functions
2 similar to be the static content, and the part that does not appear and function
3 similar to the dynamic content?

4 MR. DERGOSITS: Yeah, well, if you're reading that limitation, it
5 includes the modifier that it's the run-time view, and so, one of skill in the
6 art would only consider the need for a run-time view for dynamic aspects,
7 and not static content.

8 JUDGE HOFF: Do you have any further questions?

9 JUDGE HAHN: I have no further questions. I have no questions.

10 JUDGE KRIVAK: One second --

11 JUDGE HOFF: Yes.

12 JUDGE KRIVAK: No more questions. No. Thank you.

13 JUDGE HOFF: Thank you.

14 (Whereupon, the proceedings, at 2:01 p.m., were concluded.)

15

16

17

18

19

20

21

22

23

24

25

26